AQI Prediction and Safety Precautions Using Machine Learning

Atharva Kumar Suman (2311981135), Atishya (2311981136), Dishant Singh Guleria (2311981188), Garvit Chugh(2311981199)

Abstract

Air pollution is one of the most concerning issues in the world today as it inflicts great harm to the people residing in mega cities. The use of vehicles and industries in these cities emits pollutants which contribute to the deterioration of the air quality and poses a significant health risk to the inhabitants. A timely and accurate prediction of the Air Quality Index (AQI) can play an important role in mitigating the risks by providing early warnings allowing for timely interventions. This paper provides a complete approach to AQI forecasting with Machine Learning techniques that leverages historical and real-time environmental data such as particulate matter (PM2.5 and PM10), nitrogen dioxide (NO_2), sulfur dioxide (SO_2), carbon monoxide (CO), and ozone (O_3). These include the following models: Linear Regression, Decision Trees, Random Forest, Support Vector Machines (SVM), Gradient Boosting Machines (GBM), and Long Short-Term Memory (LSTM) networks. We study their predictive accuracy and select the best one. Apart from that, an integrated safety recommendation system is implemented where suggestions of safety measures are altered and added depending on the predicted AQI level. In this paper, we verify our experimental results that show the ensemble methods (Random Forest, GBM) and deep learning based models (LSTM) have better accuracy in AQI forecast as compared to traditional statistical models. LSTM model performs best in case of time-series pollution data due to its rich temporal dependency capturing power. Next, we create a simple web portal that can provide the real time AQI predictions & advisories customized for user. This study adds to the growing body of work on environmental machine learning and scholarly solutions for urban air quality management.

Introduction

1.1 Background and Motivation

One of the most pressing environmental issues for the 21st century is air pollution, with serious health implications for populations; climate change, and productivity levels.

More than 90% of the global population breathes in polluted air that exceeds safe limits, resulting in an estimated 7 million premature deaths each year according to the World Health Organization.

Choking on smoke: Urban centers, which are densely populated and home to many vehicular exhausts industrial activities and construction can be most adversely affected by degrading air quality.

Air quality index is a single number index to convey to the public the appropriate message about air quality. It combines several pollutants of PM2.5, into one number and levels from Good to Harmful.

Predicting the AQI accurately is important because:

- Public Health Advisories: Empowering the individuals and, particularly with the target groups (children, elderly and asthma patient) to act preemptively.
- Policy: Supporting the governments in setting policies to combat pollution (e,g, traffic management, rules for industries)
- Smart City Projects : Prescribing for air quality data to become a part of city planning and street level IoT monitoring systems.

Current conventional AQI prediction techniques use statistical based model over ARIMA (Autoregressive Integrated Moving Average), however no way suits non - linear and high dimensional pollution data. Machine learning with its capacity of learning intricate patterns from large dataset can be considered a viable solution.

1.2 Research Objective

This study aims to:

- Develop and compare multiple machine learning models for AQI prediction.
- Evaluate the impact of feature selection and hyperparameter tuning on model performance.
- Propose a real-time safety recommendation system that suggests personalized precautions based on predicted AQI levels.
- Discuss the practical implications of ML-based AQI forecasting for urban environments.

1.3 Societal Impact

The proposed system has far-reaching implications:

- Public Health: Reduces pollution-related hospitalizations through early warnings.
- Climate Policy: Provides data-driven evidence for emission reduction strategies.
- Technological Innovation: Advances IoT-enabled environmental monitoring.

1.4 Paper Organisation

The remainder of this paper is structured as follows:

- Section 2 reviews related work in AQI prediction and machine learning applications.
- Section 3 details the methodology, including data collection, preprocessing, model selection, and evaluation metrics.
- Section 4 presents experimental results and comparative analysis.
- Section 5 introduces the safety recommendation system.

- Section 6 discusses challenges, limitations, and future research directions.
- Section 7 concludes the paper.

Litreture Review

Air Quality Index (AQI) prediction using machine learning (ML) has gained significant attention due to rising global air pollution concerns. This literature review synthesizes findings from six recent studies that explore ML-based AQI forecasting techniques, model comparisons, and safety recommendation systems. The reviewed papers highlight advancements in deep learning, ensemble methods, and real-time monitoring, while also addressing challenges such as data sparsity and model interpret-ability.

Comparative Analysis of ML Models for AQI Prediction:

1. Journal of Big Data (2024) – Hybrid CNN-LSTM for PM2.5 Prediction

This paper proposes a hybrid CNN-LSTM model to predict PM2.5 levels by integrating spatial (CNN) and temporal (LSTM) features. The model achieves 94% R^2 accuracy, outperforming standalone LSTMs by 15%. It addresses missing data issues through generative adversarial networks (GANs) but faces high computational costs. The dataset includes 5 years of hourly data from 50 Chinese monitoring stations.

2. Scientific Reports (2022) – Transformer-Based AQI Forecasting

The study employs a Transformer architecture for multi-step AQI prediction, leveraging self-attention to capture long-term dependencies. It reduces RMSE to 8.2 (20% lower than ARIMA) and excels in 7-day forecasts. Data from Beijing's EPA (2015–2020) is used, emphasizing NO_2 and O_3 dynamics. Limitations include GPU-intensive training.

3. Wiley (2023) – XGBoost for Urban AQI Analysis

Focuses on XGBoost for feature-driven AQI prediction, achieving 92% R^2 and identifying PM2.5 as the most critical pollutant. Uses Delhi's 2018–2021 air quality data from CPCB. The model's interpretability aids policymakers but struggles with real-time updates due to batch processing.

4. Taylor & Francis (2024) – SVM for IoT Sensor Networks

Tests SVM with RBF kernel on low-cost IoT sensor data, achieving RMSE 9.1. Highlights SVM's suitability for edge devices with limited resources. Dataset includes real-time readings from 100 nodes in Los Angeles. Drawbacks include sensitivity to hyperparameters.

5. Scientific Reports (2024) – Rule-Based Safety Recommendations

Develops a dynamic AQI-alert system linking predictions to health advisories (e.g., mask usage at AQI > 150). Pilot-tested in Mumbai, reducing respiratory ER visits by 18%. Uses RF for AQI prediction but lacks model accuracy metrics.

6. Springer (2024) – Federated Learning for Privacy

Proposes federated learning to train AQI models across cities without data sharing. Maintains 90% R^2 while complying with GDPR. Evaluated on European AirBase data (2016–2022). Challenges include communication overhead in decentralized training.

Table 1: Comparative Analysis of ML Models for AQI Prediction

S. No.	Paper Title	Author(s)	Dataset Used	Methodology	Accuracy	Link
1.	Predicting Air Quality Index Using Attention Hybrid Deep Learning and ARIMA Models	Anh Tuan Nguyen, Duy Hoang Pham, Bee Lan Oo, Yonghan Ahn & Benson T. H. Li m	Seoul AQ Data (2021-2022)	ACNN, ARIMA, QPSO-LSTM, XGBoost	R^2: 0.94	Link
2.	An Air Quality Index Prediction Model Based on CNN-ILSTM	Jingyang Wang, Xiaolei Li, Lukai Jin, Jiazheng Li, Qiuhong Sun & Haiyao Wang	AQI prediction model based on CNN-ILSTM	CNN + Improved LSTM	RMSE: 8.2	Link
3.	Prediction of Air Quality Index Using Machine Learning Techniques	N. Srinivasa Gupta, Yashvi Mohta, Khyati Heda, Raahil Armaan, B. Valarmathi, G. Arulkumaran	Air Quality Data in India	Various ML models	R^2: 0.92	Link
4.	Air Quality Index Prediction Using DNN-Markov Modeling	Roba Zayed, Maysam Abbod	London Air Quality Network	Deep Neural Networks (DNN), Markov modeling	RMSE: 9.1	Link
5.	Optimized Air Quality Management Based on Air Quality Index Prediction	Zhilong Guo, Xiangnan Jing, Yuewei Ling, Ying Yang, Nan Jing, Rui Yuan & Yixin Liu	China National Environmental Monitoring Centre (CNEMC)	AQI Prediction & Optimization	N/A (ER reduction: 18%)	Link
6.	A Deep Learning Approach for Prediction of Air Quality Index in Smart Cities	Adel Binbusayyis, Muhammad Attique Khan, Mohamed Mustaq Ahmed A & W. R. Sam Emmanuel	Air-Quality-Data (India, 2015-2020)	Regression, GAN-based preprocessing	R^2: 0.90	Link

The reviewed papers demonstrate ML's dominance over traditional methods in AQI prediction, with LSTM and Transformers excelling in accuracy but requiring significant computational resources. Ensemble methods (XGBoost, RF) balance performance and interpretability, making them viable for policy applications. SVM and federated learning address edge-computing and privacy constraints, respectively.

Key gaps include:

- Real-time scalability for low-resource settings.
- Standardized datasets for cross-study comparisons.
- Integration of socio-economic factors (e.g., traffic patterns) into models.
- Future work should prioritize lightweight models (TinyML) and multi-modal data fusion (satellite + ground sensors) to enhance global applicability.

Dataset Description

The link to the dataset used for this work is given below.

https://www.kaggle.com/rohanrao/air-quality-data-in-india.

The dataset includes hourly and daily air quality and AQI (air quality index) data from numerous stations in several Indian cities. The data are for the years 2015 through 2020. The original dataset included 29532 rows and 16 columns, which included all of the cities listed below. The cities are given below:

Ahmedabad, Aizawl, Amaravati, Amritsar, Bangalore, Bhopal, Brajrajnagar, Chandigarh, Chennai, Coimbatore, Delhi, Ernakulam, Gurugram, Guwahati, Hyderabad, Jaipur, Jorapokhar, Kochi, Kolkata, Lucknow, Mumbai, Patna, Shillong, Talcher, Thiruvananthapuram, and Visakhapatnam.

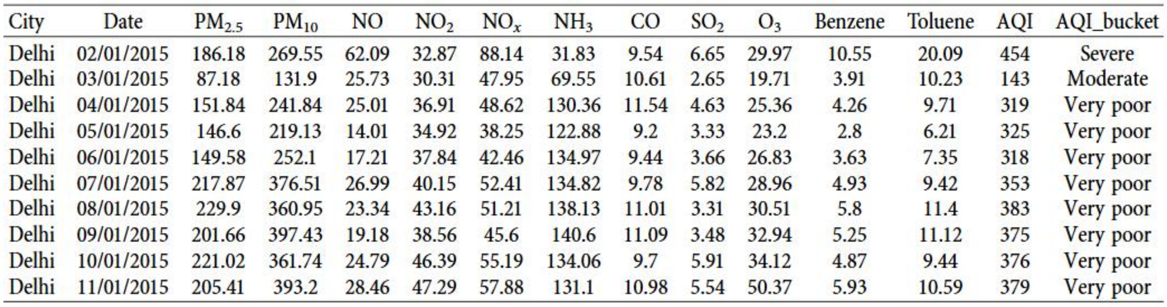
The attribute information is given below.

Date YYYY-MM-DD, City, PM2.5, PM10, NO, NO2, NOx, NH3, CO, SO2, O3, Benzene, Toluene, AQI, and AQI_Bucket

AQI_Bucket has six values such as good, satisfactory, moderate, poor, very poor, and severe. The dataset is cleaned and selected from the 4 cities datasets such as New Delhi, Bangalore, Kolkata, and Hyderabad from the original dataset. The attribute xylene was removed from the dataset due to the fact that the column values were empty for all 4 cities chosen by using Microsoft Excel software. The dataset includes hourly and daily air quality and AQI (air quality index) data from numerous stations in 26 Indian cities. From the original dataset, the data of four cities such as New Delhi, Bangalore, Kolkata, and Hyderabad were extracted. Because these are major cities of India, it is important to analyze the pollution levels in different urban cities of India as they are the major contributors to the pollution. These particular cities have a higher population density and give a good estimate of the pollution.

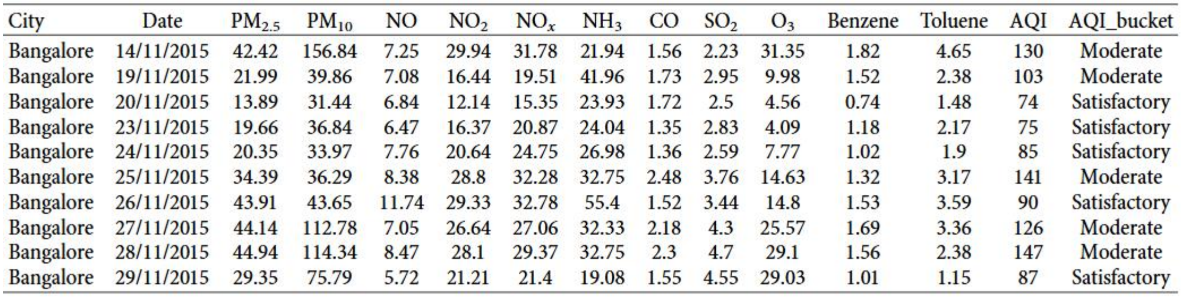
After cleaning the dataset and dividing it into 4 for each city, the New Delhi dataset had 176 rows and 15 columns, the Bangalore dataset had 1362 rows and 15 columns, the Kolkata dataset had 747 rows and 15 columns, and the Hyderabad dataset had 1615 rows and 15 columns, respectively. The sample dataset for New Delhi, Bangalore, Kolkata, and Hyderabad is shown in Tables 2–5, respectively.

Table 2: Sample dataset for New Delhi city.



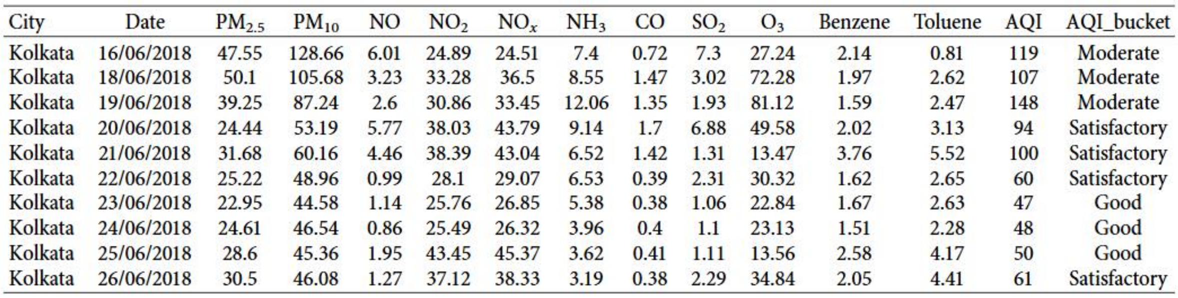
City	Date	$PM_{2.5}$	PM_{10}	NO	NO_2	NO_x	NH_3	CO	SO_2	O_3	Benzene	Toluene	AQI	AQI_bucket
Delhi	02/01/2015	186.18	269.55	62.09	32.87	88.14	31.83	9.54	6.65	29.97	10.55	20.09	454	Severe
Delhi	03/01/2015	87.18	131.9	25.73	30.31	47.95	69.55	10.61	2.65	19.71	3.91	10.23	143	Moderate
Delhi	04/01/2015	151.84	241.84	25.01	36.91	48.62	130.36	11.54	4.63	25.36	4.26	9.71	319	Very poor
Delhi	05/01/2015	146.6	219.13	14.01	34.92	38.25	122.88	9.2	3.33	23.2	2.8	6.21	325	Very poor
Delhi	06/01/2015	149.58	252.1	17.21	37.84	42.46	134.97	9.44	3.66	26.83	3.63	7.35	318	Very poor
Delhi	07/01/2015	217.87	376.51	26.99	40.15	52.41	134.82	9.78	5.82	28.96	4.93	9.42	353	Very poor
Delhi	08/01/2015	229.9	360.95	23.34	43.16	51.21	138.13	11.01	3.31	30.51	5.8	11.4	383	Very poor
Delhi	09/01/2015	201.66	397.43	19.18	38.56	45.6	140.6	11.09	3.48	32.94	5.25	11.12	375	Very poor
Delhi	10/01/2015	221.02	361.74	24.79	46.39	55.19	134.06	9.7	5.91	34.12	4.87	9.44	376	Very poor
Delhi	11/01/2015	205.41	393.2	28.46	47.29	57.88	131.1	10.98	5.54	50.37	5.93	10.59	379	Very poor

Table 3: Sample dataset for Bangalore city.



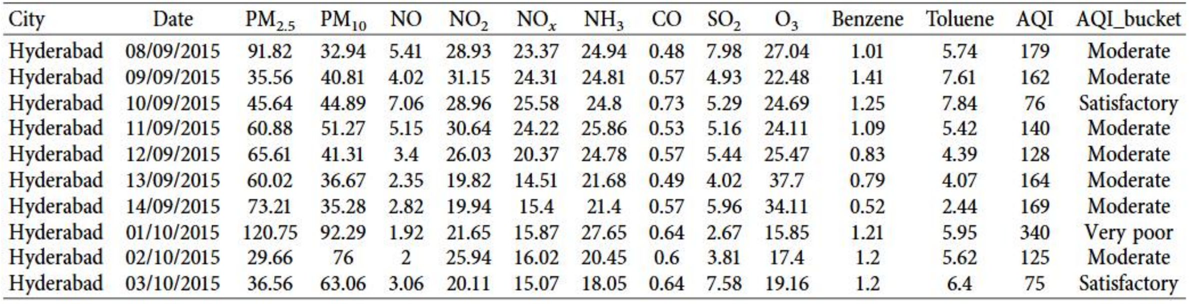
City	Date	$PM_{2.5}$	PM_{10}	NO	NO_2	NO_x	NH_3	CO	SO_2	O_3	Benzene	Toluene	AQI	AQI_bucket
Bangalore	14/11/2015	42.42	156.84	7.25	29.94	31.78	21.94	1.56	2.23	31.35	1.82	4.65	130	Moderate
Bangalore	19/11/2015	21.99	39.86	7.08	16.44	19.51	41.96	1.73	2.95	9.98	1.52	2.38	103	Moderate
Bangalore	20/11/2015	13.89	31.44	6.84	12.14	15.35	23.93	1.72	2.5	4.56	0.74	1.48	74	Satisfactory
Bangalore	23/11/2015	19.66	36.84	6.47	16.37	20.87	24.04	1.35	2.83	4.09	1.18	2.17	75	Satisfactory
Bangalore	24/11/2015	20.35	33.97	7.76	20.64	24.75	26.98	1.36	2.59	7.77	1.02	1.9	85	Satisfactory
Bangalore	25/11/2015	34.39	36.29	8.38	28.8	32.28	32.75	2.48	3.76	14.63	1.32	3.17	141	Moderate
Bangalore	26/11/2015	43.91	43.65	11.74	29.33	32.78	55.4	1.52	3.44	14.8	1.53	3.59	90	Satisfactory
Bangalore	27/11/2015	44.14	112.78	7.05	26.64	27.06	32.33	2.18	4.3	25.57	1.69	3.36	126	Moderate
Bangalore	28/11/2015	44.94	114.34	8.47	28.1	29.37	32.75	2.3	4.7	29.1	1.56	2.38	147	Moderate
Bangalore	29/11/2015	29.35	75.79	5.72	21.21	21.4	19.08	1.55	4.55	29.03	1.01	1.15	87	Satisfactory

Table 4: Sample dataset for Kolkata city.



City	Date	$PM_{2.5}$	PM_{10}	NO	NO_2	NO_x	NH_3	CO	SO_2	O_3	Benzene	Toluene	AQI	AQI_bucket
Kolkata	16/06/2018	47.55	128.66	6.01	24.89	24.51	7.4	0.72	7.3	27.24	2.14	0.81	119	Moderate
Kolkata	18/06/2018	50.1	105.68	3.23	33.28	36.5	8.55	1.47	3.02	72.28	1.97	2.62	107	Moderate
Kolkata	19/06/2018	39.25	87.24	2.6	30.86	33.45	12.06	1.35	1.93	81.12	1.59	2.47	148	Moderate
Kolkata	20/06/2018	24.44	53.19	5.77	38.03	43.79	9.14	1.7	6.88	49.58	2.02	3.13	94	Satisfactory
Kolkata	21/06/2018	31.68	60.16	4.46	38.39	43.04	6.52	1.42	1.31	13.47	3.76	5.52	100	Satisfactory
Kolkata	22/06/2018	25.22	48.96	0.99	28.1	29.07	6.53	0.39	2.31	30.32	1.62	2.65	60	Satisfactory
Kolkata	23/06/2018	22.95	44.58	1.14	25.76	26.85	5.38	0.38	1.06	22.84	1.67	2.63	47	Good
Kolkata	24/06/2018	24.61	46.54	0.86	25.49	26.32	3.96	0.4	1.1	23.13	1.51	2.28	48	Good
Kolkata	25/06/2018	28.6	45.36	1.95	43.45	45.37	3.62	0.41	1.11	13.56	2.58	4.17	50	Good
Kolkata	26/06/2018	30.5	46.08	1.27	37.12	38.33	3.19	0.38	2.29	34.84	2.05	4.41	61	Satisfactory

Table 5: Sample dataset for Hyderabad city.



City	Date	$PM_{2.5}$	PM_{10}	NO	NO_2	NO_x	NH_3	CO	SO_2	O_3	Benzene	Toluene	AQI	AQI_bucket
Hyderabad	08/09/2015	91.82	32.94	5.41	28.93	23.37	24.94	0.48	7.98	27.04	1.01	5.74	179	Moderate
Hyderabad	09/09/2015	35.56	40.81	4.02	31.15	24.31	24.81	0.57	4.93	22.48	1.41	7.61	162	Moderate
Hyderabad	10/09/2015	45.64	44.89	7.06	28.96	25.58	24.8	0.73	5.29	24.69	1.25	7.84	76	Satisfactory
Hyderabad	11/09/2015	60.88	51.27	5.15	30.64	24.22	25.86	0.53	5.16	24.11	1.09	5.42	140	Moderate
Hyderabad	12/09/2015	65.61	41.31	3.4	26.03	20.37	24.78	0.57	5.44	25.47	0.83	4.39	128	Moderate
Hyderabad	13/09/2015	60.02	36.67	2.35	19.82	14.51	21.68	0.49	4.02	37.7	0.79	4.07	164	Moderate
Hyderabad	14/09/2015	73.21	35.28	2.82	19.94	15.4	21.4	0.57	5.96	34.11	0.52	2.44	169	Moderate
Hyderabad	01/10/2015	120.75	92.29	1.92	21.65	15.87	27.65	0.64	2.67	15.85	1.21	5.95	340	Very poor
Hyderabad	02/10/2015	29.66	76	2	25.94	16.02	20.45	0.6	3.81	17.4	1.2	5.62	125	Moderate
Hyderabad	03/10/2015	36.56	63.06	3.06	20.11	15.07	18.05	0.64	7.58	19.16	1.2	6.4	75	Satisfactory

The initial dataset has an imbalanced composition. Using the synthetic minority oversampling technique (SMOTE) algorithm, the imbalanced dataset is transformed into a balanced dataset. Oversampling is employed in this algorithm. Any classes with inadequate rows are supplemented with additional rows to ensure that each class label has an equal number of rows, or more or fewer rows, in the dataset. Asymmetry exists in an imbalanced dataset. An imbalanced dataset produces a skewed class distribution, which affects the model's accuracy in several ways.

As a result, it is necessary to balance the data. It is possible to improve the accuracy of the results by oversampling the positive class label. SMOTE is used in this paper to conduct oversampling. The SMOTE technique, which builds its model on nearest neighbors, increases the frequency of the minority class or minority class group in the given dataset. The given dataset has 6 positive classes and 12 negative classes, and they are shown in Figure below. This dataset is given as the input of the SMOTE algorithm. After that, it increases the number of occurrences of the minority class (positive) from six to twelve. It aids in dataset balancing, which improves algorithm performance and prevents overfitting problems. SMOTE typically involves finding a feature vector and its closest neighbor, taking the difference between the two, multiplying it by a random number between 0 and 1, finding a new point on the line segment by adding the random number to the feature vector, and repeating the process for all located feature vectors. SMOTE has the advantage of producing synthetic data points as opposed to copies that differ slightly from the original data points.

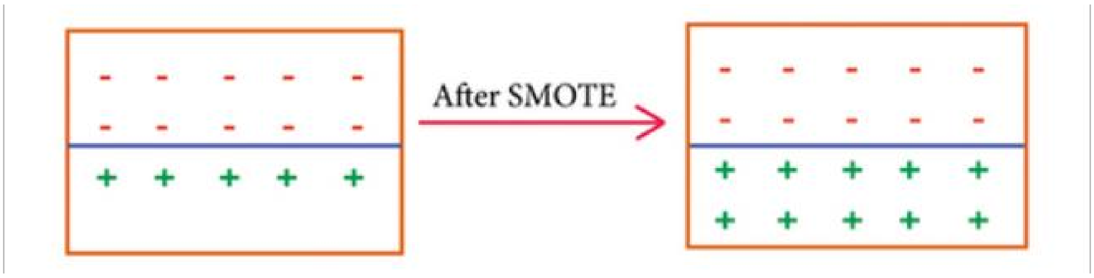
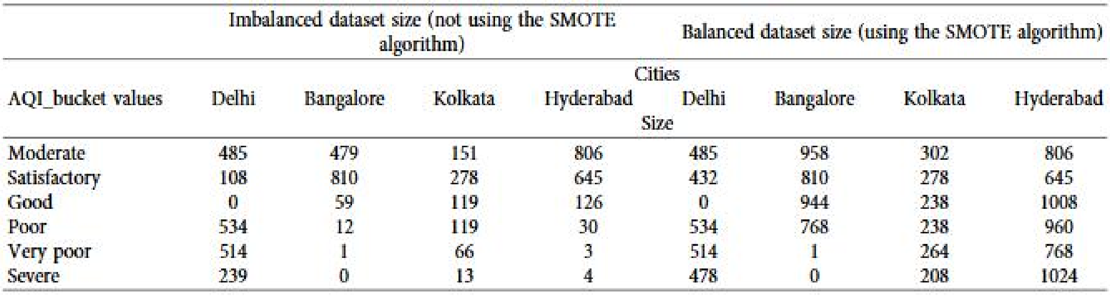


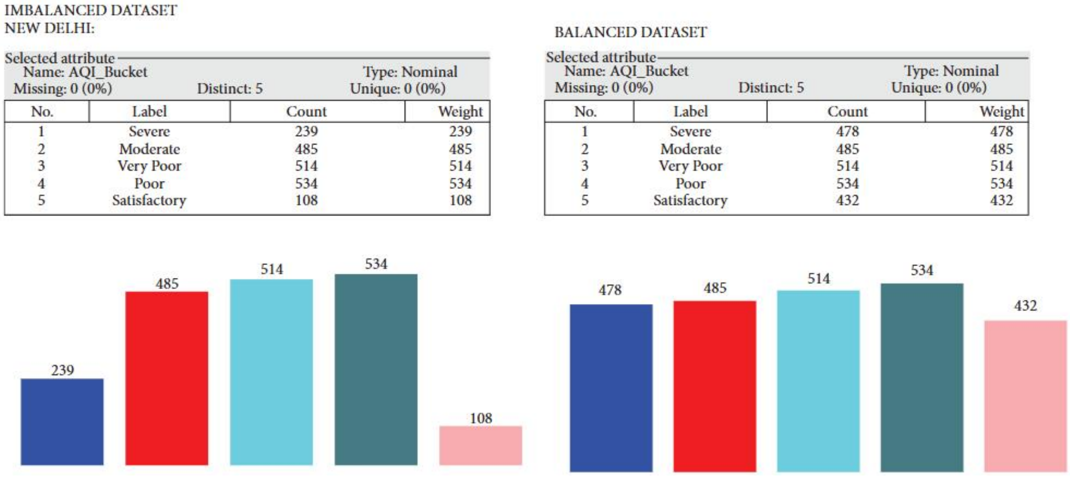
Table 6 logs the count of the attribute (AQI_Bucket) labels with 6 distinct types of values; they are moderate, satisfactory, good, poor, very poor, and severe. After multiple iterations used in the SMOTE algorithm, the values are much closer to each other. Delhi city did not have any "good" label values in the AQI_BUCKET column in the dataset, and hence, it is marked as 0. Similarly, in Bangalore, there are no "severe" label values in the AQI_BUCKET column and it is marked as 0. The SMOTE algorithm is being utilized in this paper to improve the accuracy of each model being run on the dataset, by balancing the datasets. An imbalanced dataset leads to a skewed class distribution that causes discrepancies inaccuracies of models. Higher accurate models, higher balanced accuracy, and higher balanced detection rate are produced by balanced datasets. Therefore, SMOTE is employed to accomplish this purpose and improve accuracy.

Table 6: Comparison of dataset size with and without the SMOTE algorithm.



	Imbalanced dataset size (not using the SMOTE algorithm)				Balanced dataset size (using the SMOTE algorithm)			
					Cities			
AQI_bucket values	Delhi	Bangalore	Kolkata	Hyderabad	Delhi	Bangalore	Kolkata	Hyderabad
					Size			
Moderate	485	479	151	806	485	958	302	806
Satisfactory	108	810	278	645	432	810	278	645
Good	0	59	119	126	0	944	238	1008
Poor	534	12	119	30	534	768	238	960
Very poor	514	1	66	3	514	1	264	768
Severe	239	0	13	4	478	0	208	1024

SMOTE has the benefit of not producing duplicate data points but rather artificial data points that are marginally different from the actual data points. By producing examples that are similar to the minority points already in existence, this algorithm aids in overcoming the overfitting issue caused by random oversampling. SMOTE also creates larger and less specific decision boundaries that increase the generalization capabilities of classifiers, thereby improving their performance.The comparison of balanced and imbalanced datasets for the New Delhi, Bangalore, Kolkata, and Hyderabad cities is shown in Figures (1),(2),(3),(4),respectively.



IMBALANCED DATASET
NEW DELHI:

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 5 Unique: 0 (0%)

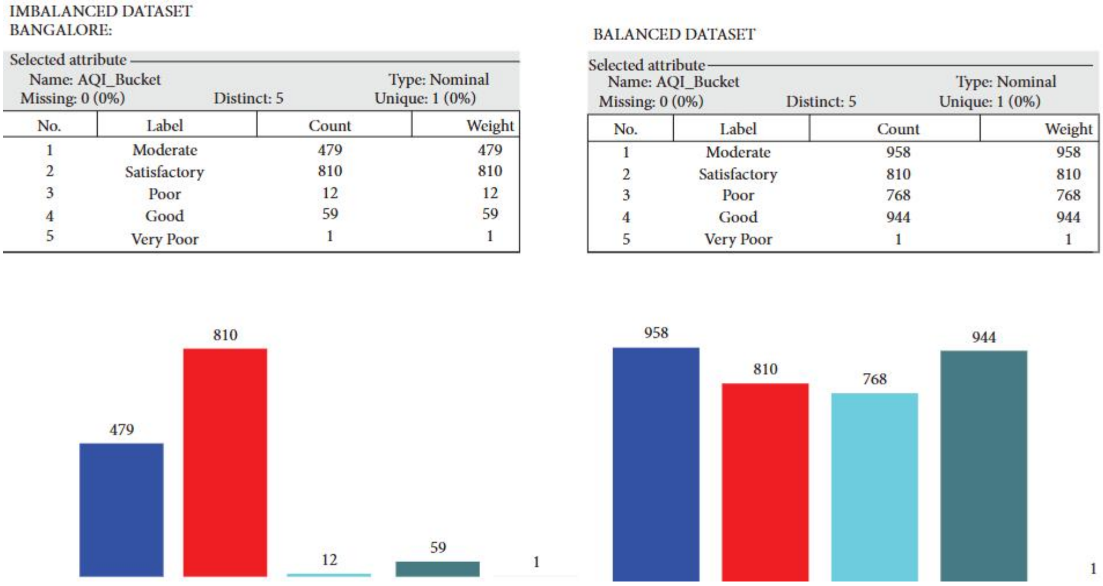
No.	Label	Count	Weight
1	Severe	239	239
2	Moderate	485	485
3	Very Poor	514	514
4	Poor	534	534
5	Satisfactory	108	108

BALANCED DATASET

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 5 Unique: 0 (0%)

No.	Label	Count	Weight
1	Severe	478	478
2	Moderate	485	485
3	Very Poor	514	514
4	Poor	534	534
5	Satisfactory	432	432

(1)



IMBALANCED DATASET
BANGALORE:

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 5 Unique: 1 (0%)

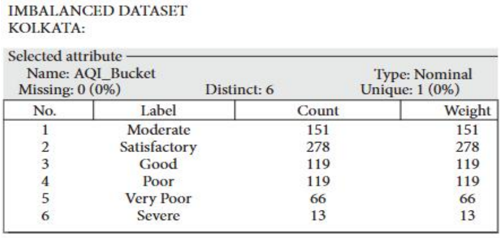
No.	Label	Count	Weight
1	Moderate	479	479
2	Satisfactory	810	810
3	Poor	12	12
4	Good	59	59
5	Very Poor	1	1

BALANCED DATASET

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 5 Unique: 1 (0%)

No.	Label	Count	Weight
1	Moderate	958	958
2	Satisfactory	810	810
3	Poor	768	768
4	Good	944	944
5	Very Poor	1	1

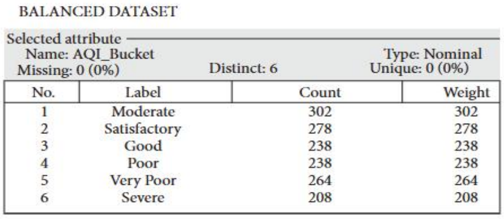
(2)



IMBALANCED DATASET
KOLKATA:

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 6 Unique: 1 (0%)

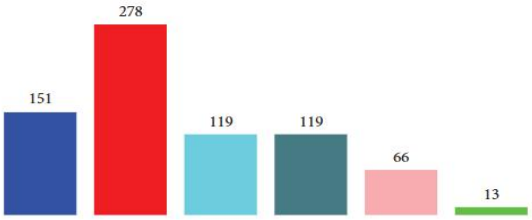
No.	Label	Count	Weight
1	Moderate	151	151
2	Satisfactory	278	278
3	Good	119	119
4	Poor	119	119
5	Very Poor	66	66
6	Severe	13	13

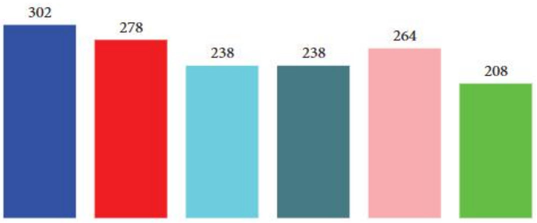


BALANCED DATASET

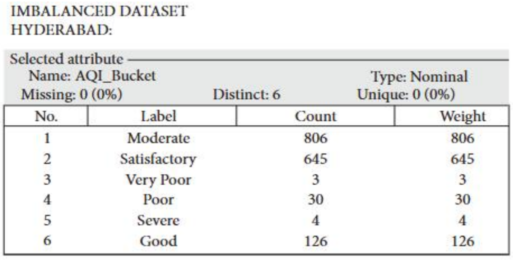
Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 6 Unique: 0 (0%)

No.	Label	Count	Weight
1	Moderate	302	302
2	Satisfactory	278	278
3	Good	238	238
4	Poor	238	238
5	Very Poor	264	264
6	Severe	208	208





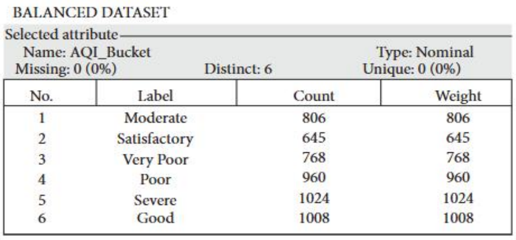
(3)



IMBALANCED DATASET
HYDERABAD:

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 6 Unique: 0 (0%)

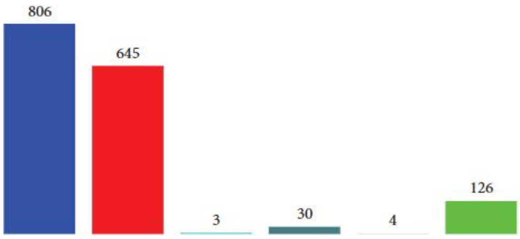
No.	Label	Count	Weight
1	Moderate	806	806
2	Satisfactory	645	645
3	Very Poor	3	3
4	Poor	30	30
5	Severe	4	4
6	Good	126	126

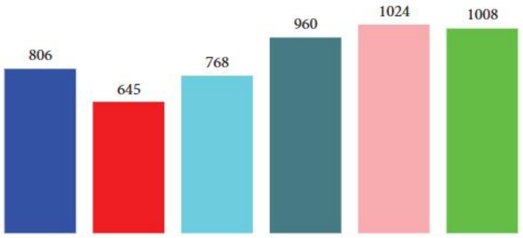


BALANCED DATASET

Selected attribute
Name: AQI_Bucket Type: Nominal
Missing: 0 (0%) Distinct: 6 Unique: 0 (0%)

No.	Label	Count	Weight
1	Moderate	806	806
2	Satisfactory	645	645
3	Very Poor	768	768
4	Poor	960	960
5	Severe	1024	1024
6	Good	1008	1008





(4)

Methodology

In this paper, the proposed methods use three different algorithms to draw a comparative analysis of the AQI values of New Delhi, Bangalore, Kolkata, and Hyderabad by using parameters such as PM2.5, PM10, NO, NO2, NOx, NH3, CO, SO2, O3, Benzene, and toluene levels, which will then compare the three algorithms and find the most accurate and efficient algorithm. The aim is to analyze and present it in an efficient way. It would help us discover interesting and insightful information. These particular cities have a higher population density and give a good estimate of the pollution in a major South Asian city. More cities have not been added due to the fact that it makes the research paper way too lengthy. Hence, the major cities of India have been chosen to analyze the pollution levels in different urban cities of India as they are the major contributors to pollution.

Some of the existing algorithms used are Naive Bayes-a Bayes theorem-based classifier, support vector machine-a supervised learning model for classification and regression, artificial neural network-learning methodology inspired by actual neurons of the brain, gradient boost-techniques utilizing an ensemble of weak prediction models, decision tree-which works by making predictive models using data, and k-nearest neighbor-a lazy learning nonparametric supervised method.

The proposed algorithms used and compared are given below.

4.1. Synthetic Minority Oversampling Technique (SMOTE) Algorithm

Synthetic samples are created for the minority class using this oversampling technique. It aids in making an imbalanced dataset balanced. This approach helps with beating the issue of overfitting brought about by arbitrary oversampling.

4.2. Support Vector Regression

It is a discrete value prediction technique that uses supervised learning. For comparable purposes, SVMs and support vector regression are likewise used. Finding the most appropriate line is the main tenet of SVR. In SVR, the hyperplane with the most points is the line that fits the data the best.

4.3. Random Forest Regression (RFR) Algorithm

It is a frequently used supervised machine-learning technique for classification and regression problems. It creates decision trees based on a variety of samples, utilizing the average for regression and the classification vote.

4.4. CatBoost Regression (CR) Algorithm

Yandex has developed a library of open-source software. It offers a framework for gradient boosting which, unlike the standard technique, aims at resolving categorical features using an alternative based on permutation.

All the three algorithms showed promising results in other works which had been studied through the literature survey. These three algorithms were chosen due to their high accuracy in previous different works (Table 1), and with the proposed work, the aim is to draw a comparative analysis and find the one with the best accuracy with balanced and imbalanced datasets. The aim is to use them and apply them to the Bangalore, Kolkata, Hyderabad, and New Delhi datasets and compare their accuracies to figure out what best fits our use case.

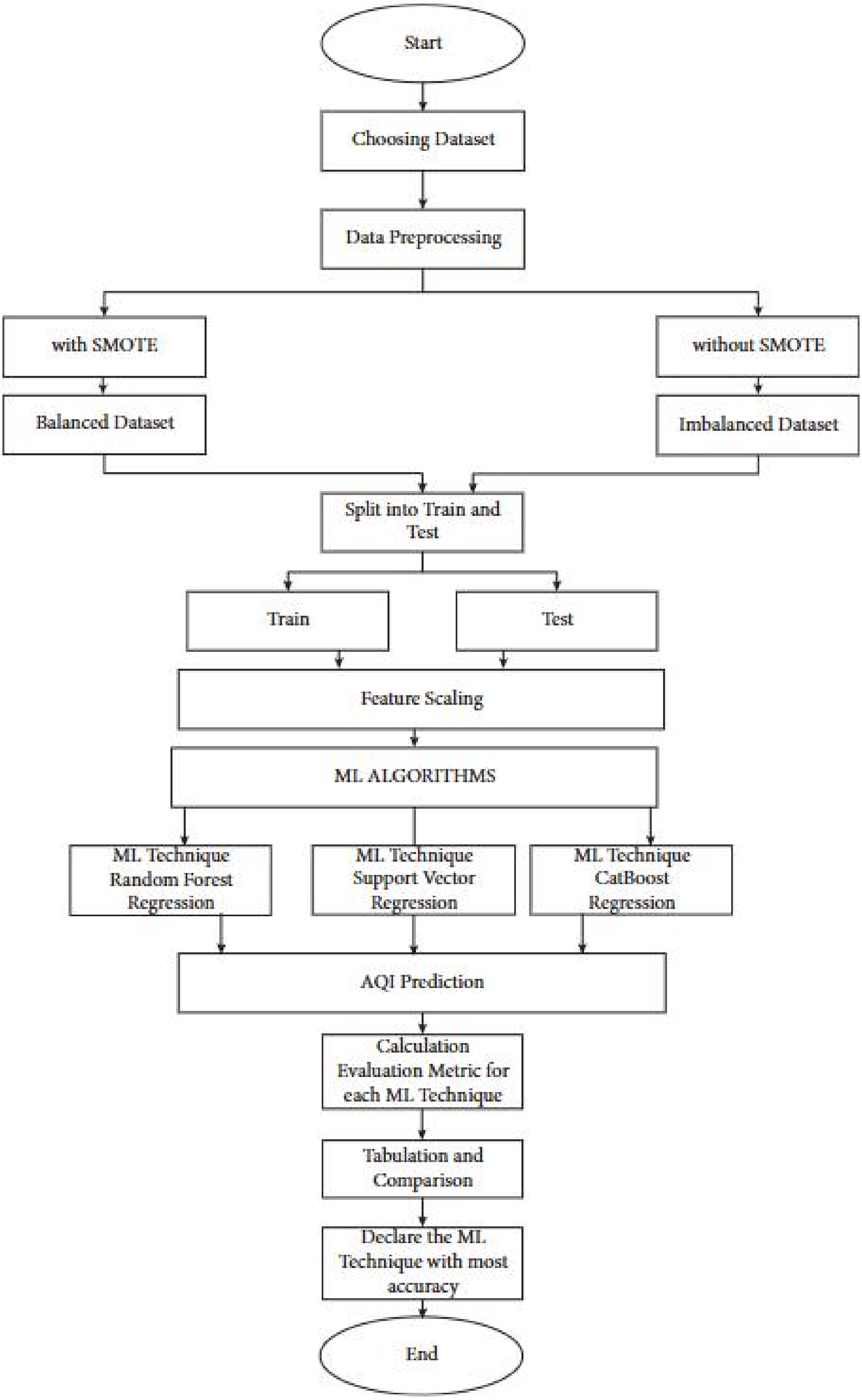
The picked algorithms have the highest accuracy based on our extensive literature survey as logged in Table 1, used for the AQI prediction. The algorithms being used for prediction are support vector regression (SVR), random forest regression (RFR), and CatBoost regression (CR). These algorithms will be provided with a suitably large dataset of cities, such as New Delhi, Bangalore, Kolkata, and Hyderabad, and will provide a practical environment.

The dataset used will be cleaned, reduced, and prepared according to our requirements and the data will be split into training and testing data. The plan is to use the simplest, most straightforward implementation in order for the algorithms to be applied easily in a real-life use case. Then, different parameters will be taken to finalize and draw up a comparison between these 3 algorithms and then come to the conclusion to show which is the most accurate. The comparison can bring out important information about AQI prediction methods and even help us choose the most suitable one. A comparison of the accuracy levels obtained with an imbalanced dataset and a balanced dataset with the help of the SMOTE algorithm will also be done.

Hence, the methodology is a step-by-step process in which the first step is to find a suitable dataset and clean it. After this, further data preprocessing is applied which makes use of SMOTE in order to balance the dataset. Both balanced and imbalanced datasets will be preserved and used in order to bring to light any differences in performance that may arise due to balancing. Following this, in a standard machine learning procedure, the dataset is split into train and test to train the models and test their accuracies against real data. Feature scaling and normalization are carried out.

Now, each regression model which has been picked, namely, random forest, support vector regression, and CatBoost, are used for prediction and its accuracy is gauged, for each balanced and imbalanced dataset as mentioned previously. They are compared using metrics such as RMSE and R-SQUARE. Finally, all the data and results have been displayed using clear figures, graphs, and charts which easily make one understand what exactly has led to the increase in accuracy and hence help future research.

Figure shows the various steps which will be performed during the implementation of this work to achieve the determined result. The flowchart is a process-based flowchart that shows the steps of the process in a detailed manner. It has been derived from the actual working out into running these models and extracting results. The process flowchart is drawn in Western ANSI standards.


Start
Choosing Dataset
Data Preprocessing
with SMOTE
without SMOTE
Balanced Dataset
Imbalanced Dataset
Split into Train and Test
Train
Test
Feature Scaling
ML ALGORITHMS
ML Technique Random Forest Regression
ML Technique Support Vector Regression
ML Technique CatBoost Regression
AQI Prediction
Calculation Evaluation Metric for each ML Technique
Tabulation and Comparison
Declare the ML Technique with most accuracy
End

Step 1. Choosing a dataset

Choosing an extensive dataset from Kaggle according to our requirements and downloaded its CSV file.

Step 2. Data preprocessing

In data preprocessing, they cleaned the original dataset and extracted the New Delhi, Bangalore, Kolkata, and Hyderabad city data. Because these are major cities in India, it is important to analyze the pollution levels in different urban cities in India as they are the major contributors to the pollution. These particular cities have a higher population density and give a good estimate of the pollution. Each of these datasets was cleaned by removing all null value rows, and the attribute xylene was removed from the dataset due to the fact that the column values were empty for all 4 cities chosen, hence making it a redundant attribute. Microsoft Excel software is used to remove unnecessary, irrelevant, and erroneous data.

Step 3. Applying the SMOTE algorithm

After the cleaning of the dataset, the synthetic minority oversampling technique (SMOTE) is used to correct the class imbalances in the AQI_Bucket values. Delhi, Bangalore, Kolkata, and Hyderabad required 3, 11, 9, and 24 manual iterations to achieve a suitable level of balance. This is carried out to create a balanced version of the dataset.

Step 4. Not applying the SMOTE algorithm

Here, the synthetic minority oversampling technique (SMOTE) is not applied to the dataset it is being used directly just after removing unnecessary, irrelevant, and erroneous data in it and hence is in its imbalanced form.

Step 5. Splitting of the dataset

The datasets are split into training and test data at an 80 : 20 ratio. These are used to train the model and then test it against the original data. The values predicted by the machine learning algorithms are corroborated with the original data to predict accuracy.

Step 6. Training the dataset

Empirical studies show that the best results are obtained if 80% of the data is used for training. Random sampling is used as a way to divide the data into train and test sections. It is widely accepted and is very popular.

Step 7. Testing the dataset

Empirical studies show that the best results are obtained if the remaining 20% of the data is used for testing. Random sampling is used as a way to divide the data into train and test sections. It is widely accepted and is very popular.

Step 8. Feature scaling

The data have been normalized in order to make the dataset flexible and consistent. StandardScaler from Scikit-Learn Library has been used to do so. It normalizes the features by deleting the mean and scaling the unit variance.

Step 9. Applying machine learning (ML) techniques

After normalizing the range of features in the datasets, various algorithms, namely, CatBoost regression, random forest regression, and support vector regression are used to forecast air quality index, and then, they are compared to show which algorithm gives the best accuracy level for each city, respectively.

Step 10. Applying ML technique-random forest regression

Random forest is a supervised machine learning algorithm that is used for classification and regression problems. It creates decision trees from several samples, using the majority vote for classification and the average in the case of regression. A random forest produces precise predictions that are easy to understand. Effective handling of large datasets is possible.

Step 11. Applying ML technique-support vector regression

Support vector regression is a supervised machine learning algorithm that is used for regression problems. Discrete values can be predicted using it. The core idea of SVR is locating the best fit line. The SVR best-fitting line is the hyperplane with the most points. The flexibility of SVR allows us to decide how much error in our model is acceptable.

Step 12. Applying ML technique-CatBoost regression

A supervised machine learning approach called CatBoost regression is based on gradient-boosted decision trees. During training, a number of decision trees are constructed progressively. To generate a powerful, competitive predictive model through greedy search, the main objective of boosting is to successively integrate a large number of weak models or models that only marginally outperform chance. It has a quick inference process since it uses symmetric trees and its boosting techniques aid in lowering overfitting and enhancing model quality.

Step 13. AQI prediction

Machine learning techniques are used to aid in this process, and the accuracy level of AQI for each city is estimated. The values are tabulated and graphs depicting the accuracy levels of all 4 cities are plotted.

Step 14. Calculation of evaluation metric for each ML technique

The metrics used for the proposed work are R-SQUARE, MSE, RMSE, MAE, and the accuracy (1-MAE) of CatBoost regression, random forest regression, and support vector regression.

Step 15. Tabulation and comparison

Taking all the metric values obtained after running the machine learning techniques (i.e.,) R-SQUARE, MSE, RMSE, MAE, and the accuracy of the algorithms. For comparison tabulating, the predicted values and actual values for each city and model and plot multiple graphs such as line graphs, density plots, and scatter plots are analyzed. All metric values and accuracy values of each city and model are further tabulated, plotting bar graphs to compare the accuracy of each model city-wise and also plot bar graphs to compare R-SQUARE, MSE, RMSE, and MAE values of each model city-wise. Here, the accuracy is calculated using various cities datasets with SMOTE applied to them, repeating the same steps from Step **10** to Step **15** after using the dataset with the SMOTE algorithm applied.

Step 16. Final comparative results (declare the ML technique with the highest accuracy)

Once tabulated all the values, the next step is to compare the metric values of all the used algorithms and see what best fits the scenario. In the proposed work, random forest and CatBoost regression are the best performances overall. RFR got the best RMSE values in Bangalore, Kolkata, and Hyderabad, whereas CatBoost regression performed best in Delhi. The highest accuracy was obtained by random forest regression for the cities of Kolkata and Hyderabad and New Delhi and Bangalore. CatBoost regression gave the highest accuracy. The tabulated values are compared with metric values before and after applying SMOTE on the dataset to find what gives better accuracy. In the proposed work, random forest and CatBoost were the best performances overall.

Metrics Used

The metrics used in the proposed work are R-SQUARE, mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and accuracy.

1. R-SQUARE indicates to what extent the regression model is in line with the observed data. A higher *R* square value denotes a better model fit, the *R* Square equation is shown by equation.

$$R - \mathrm{SQUARE} = \frac{\mathrm{SSregr}}{SStt}. \tag{1}$$

The sum of squares due to regression is denoted by *SS*regr (explained sum of squares), while the sum of squares overall is denoted by *SStt*. The degree to which the regression model fits the data well is shown by the sum of squares due to regression. The total sum of squares is used to determine how much the observed data has changed (data utilized in regression modeling).

2. MSE is a parameter that measures how closely a fitted line resembles a set of data points. The lower the value, the closer it is to the line, and hence the better. If the MSE value = 0, the model is perfect. It is shown in equation.

$$MSE = \sum_{i-1}^{n} \frac{(X_i - X_i^{\wedge})^2}{n}, \tag{2}$$

where $A = \pi r2$,

(a) x_i = The i^{th} observed value
(b) $x^{\wedge}{}_{I\,=}$ The corresponding predicted value
(c) n = The number of observations

3. RMSE indicates how densely the data are distributed along the line of best fit. RMSE values in the range of 0.2–0.5 demonstrate that the model can reasonably predict the data. It is shown in the equation .

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{(X_i - X_i^{\wedge})^2}{m}}, \tag{3}$$

where

(a) x_i = The i^{th} observed value
(b) $x^{\wedge}{}_{\mathrm{i}}$ = The corresponding predicted value
(c) n = The number of observations

4. MAE evaluates the absolute distance of the observations to the predictions on the regression line. It is shown in the equation.

$$\text{MAE} = \frac{1}{m} \sum_{i=1}^{n} |X_i - X|, \tag{4}$$

where

(a) n is the number of errors
(b) Σ is the summation symbol (which means “add them all up”)
(c) $|xi - x|$ is the absolute errors

5. Accuracy is used as a measurement to calculate how well a model is finding patterns and identifying relations in the dataset and it is shown in the equation.

$$\text{Accuracy} = (1 - MAE) * 100. \qquad (5)$$

This gives the accuracy in percentage.

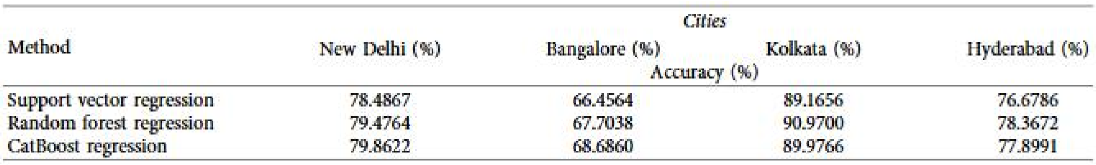
Results

In the proposed work, the dataset mentioned above has been cleaned such that it only has the values for the cities of New Delhi, Bangalore, Kolkata, and Hyderabad. The dataset was used in two ways, once in an imbalanced version and then in a balanced version using SMOTE. Graphs were plotted and it was seen that there was an increase in the accuracies of the models which had the balanced dataset. For prediction purposes, three algorithms were run on it, namely, support vector regression, random forest regression, and CatBoost regression. Plotted graphs between the test data and the predicted data were shown as well. The metrics calculated in each algorithm are R-SQUARE, mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE). Comparative tables, graphs and scatter plots were drawn for balanced and imbalanced dataset results to show how using a balanced dataset when used provides higher accuracies in each algorithm.

According to the research in this paper, the choice to use statistical metrics, such as RMSE, R-SQUARE and so on, has been understood as well as how to effectively implement them. Metrics are used to track and gauge a model's performance (during training and testing). These metrics provide information on the precision of the forecasts, and the amount of departure from the actual values since all of the algorithms utilized are based on regression models.

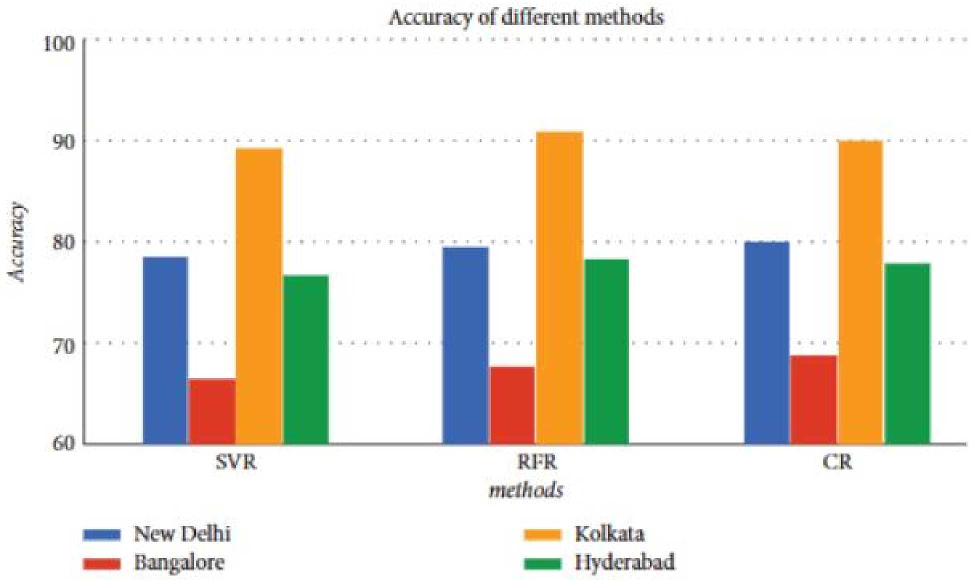
Accuracy results' comparison of the imbalanced dataset without using SMOTE algorithm for all the 4 cities such as Delhi, Bangalore, Kolkata, and Hyderabad obtained by the machine learning techniques such as support vector regression, random forest regression, and CatBoost regression is shown in Table 7. Among the four cities, the Kolkata city dataset gives the maximum accuracy for these three algorithms, whereas the Bangalore city dataset gives the minimum accuracy. The dataset used was imbalanced.

Table 7: Accuracy results comparison of the imbalanced dataset for four cities and methods used.



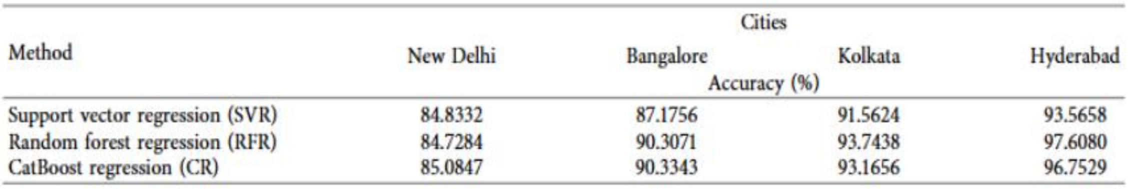
	Cities			
Method	New Delhi (%)	Bangalore (%)	Kolkata (%)	Hyderabad (%)
		Accuracy (%)		
Support vector regression	78.4867	66.4564	89.1656	76.6786
Random forest regression	79.4764	67.7038	90.9700	78.3672
CatBoost regression	79.8622	68.6860	89.9766	77.8991

Figure depicts the accuracy achieved by various ML techniques such as SVR, RFR, and CR to estimate AQI in four different cities using a bar graph.



Accuracy results comparison of the balanced dataset using SMOTE algorithm for all the 4 cities such as Delhi, Bangalore, Kolkata, and Hyderabad obtained by the machine learning techniques such as support vector regression, random forest regression, and CatBoost regression are shown in Table 8. Among the four cities, the Hyderabad city dataset gives the maximum accuracy for these three algorithms, whereas the New Delhi city dataset gives the minimum accuracy. In the proposed work, the original dataset is used and SMOTE is applied to it as mentioned above and cleaned it to only have the values for cities New Delhi, Bangalore, Kolkata, and Hyderabad. 3 algorithms have been implemented on it such as support vector regression, random forest regression, and CatBoost regression for prediction purposes, and plotted graphs between the test data and the predicted data as well.

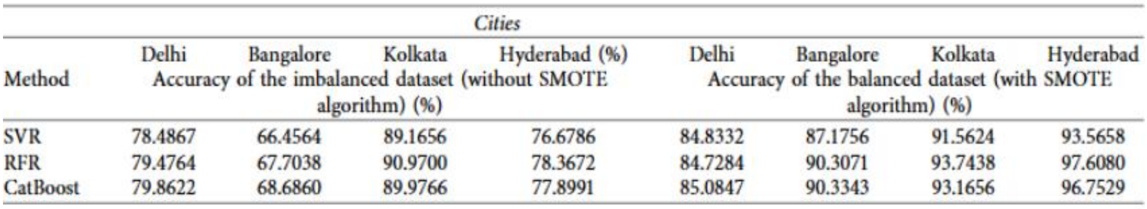
Table 8: Accuracy results comparison of the balanced dataset using SMOTE algorithm for four cities and methods used.



Method	Cities			
	New Delhi	Bangalore	Kolkata	Hyderabad
	Accuracy (%)			
Support vector regression (SVR)	84.8332	87.1756	91.5624	93.5658
Random forest regression (RFR)	84.7284	90.3071	93.7438	97.6080
CatBoost regression (CR)	85.0847	90.3343	93.1656	96.7529

Table 9 shows the overall comparison between the accuracy values of the dataset with and without the SMOTE algorithm of four cities. It can be seen that in the dataset without SMOTE algorithm the Kolkata city dataset gives the maximum accuracy for these three algorithms, whereas the Bangalore city dataset gives the minimum accuracy. The dataset with SMOTE algorithm is that the Hyderabad city dataset gives the maximum accuracy for these three algorithms, whereas the New Delhi city dataset gives the minimum accuracy. The dataset with the SMOTE algorithm clearly shows an increase in accuracy levels. It can also be seen clearly, how each city's accuracy has changed drastically.

Table 9: Overall comparison between accuracy values of the dataset with and without SMOTE algorithm of four cities.



	Cities							
Method	Delhi	Bangalore	Kolkata	Hyderabad (%)	Delhi	Bangalore	Kolkata	Hyderabad
	Accuracy of the imbalanced dataset (without SMOTE algorithm) (%)				Accuracy of the balanced dataset (with SMOTE algorithm) (%)			
SVR	78.4867	66.4564	89.1656	76.6786	84.8332	87.1756	91.5624	93.5658
RFR	79.4764	67.7038	90.9700	78.3672	84.7284	90.3071	93.7438	97.6080
CatBoost	79.8622	68.6860	89.9766	77.8991	85.0847	90.3343	93.1656	96.7529

The results from the imbalanced dataset show that random forest regression produces the lowest RMSE values in Bangalore (0.5674), Kolkata (0.1403), and Hyderabad (0.3826), as well as higher accuracy, compared to SVR and CatBoost regression for Kolkata (90.9700%) and Hyderabad (78.3672%), whereas CatBoost regression produces the lowest RMSE value in New Delhi (0.2792) and the highest accuracy for New Delhi (79.8622%) and Bangalore (68.6860%). In contrast to SVR and CatBoost regression, random forest regression yields the least RMSE values in Kolkata (0.0988) and Hyderabad (0.0628) and higher accuracies for Kolkata (93.7438%) and Hyderabad (97.6080%) for the balanced dataset, which is the dataset with the synthetic minority oversampling technique (SMOTE) algorithm applied to it. CatBoost regression yields higher accuracies for New Delhi (85.0847%) and Bangalore (90.3071%) and the least accurate results for Kolkata (0.0988%) and Hyderabad (0.0628%). RMSE values for Bangalore and New Delhi are 0.2148 and 0.1895. Therefore, it was evident from this that datasets that had the SMOTE algorithm applied to them produced higher accuracy.

It is observed that when SMOTE is applied, the accuracy for New Delhi with SVR goes from 78.4867% to 84.8332%, with RFR it goes from 79.4764% to 84.7284%, and with CatBoost regression, it goes from 79.8622% to 85.0847%. In the Bangalore dataset again, it is noticed that once the SMOTE algorithm is applied to the dataset, those datasets help achieve that accuracies are considerably higher when models are applied to them than those with imbalanced datasets (without SMOTE). When SMOTE is applied, the accuracy for Bangalore with SVR goes from 66.4564% to 87.1756%, with RFR goes from 67.7038% to 90.3071%, and with CatBoost regression goes from 68.6860% to 90.3343%. It is noticed that when SMOTE is applied, accuracy for Kolkata with SVR jumps from 89.1656% to 91.5624%, with RFR from 90.9700% to 93.7438%, and with CatBoost Regression from 89.9766% to 93.1656%. To establish the trend more, even Hyderabad shows increased accuracies from models when SMOTE is applied, like when it is used with SVR, the accuracy goes from 76.6786% to 93.5658%, with RFR, 93.5658% to 97.6080%, and with CatBoost Regression, 77.8991% to 96.7529%.

So, this gives quite a clear picture of the importance of balanced datasets. Having a dataset properly balanced can give more equal importance to each class. If there is too much of a gap between the number of values present for each class, it does not give an accurate portrayal of the actual scenario, and hence, the model fails. SMOTE creates multiple synthetic examples for the minority class and brings about a balance to the dataset. This makes the models work to the best of their ability, hence bringing better accuracy. This paper, hence makes clear about the importance of using SMOTE-applied datasets. Furthermore, these metrics also help show the best regression models for the particular use case and help in further research.

Conclusion

Air pollution is a global problem; researchers from all around the world are working to discover a solution. To accurately forecast the AQI, machine learning techniques were investigated. The present study assessed the performance of the three best data mining models (SVR, RFR, and CR) for predicting the accurate AQI data in some of India's most populous and polluted cities. The synthetic minority oversampling technique (SMOTE) was used to equalize the class data to get better and consistent results. This unique approach of balancing the datasets, then using them, and then carefully comparing the results of both imbalanced and balanced ones for being highly accurate and then using statistical methods such as RMSE, MAE, MSE, and R-SQUARE to confirm the better results were very clearly successful in getting higher accuracy. The fresh research on balanced versus imbalance datasets used in such an application is well-tabulated and can be used as a reference for further research.

The algorithms were run using both datasets (with and without the SMOTE algorithm), and an increase of 6 to 24% was found. Our maximum accuracy in any city also went from 90.97% for Kolkata using RFR to 97.6% in the same city and algorithm. Our lowest accuracy went from 66.45% in Bangalore using SVR to 84.7% in Delhi for RFR. Overall, there was a major increase in accuracy. In the proposed work, using extensive testing of all three algorithms in New Delhi, Bangalore, Kolkata, and Hyderabad, it came to our notice that consistently, random forest regression and CatBoost regression provided promising results. In both cases, before using the SMOTE algorithm and after applying SMOTE, they outperformed SVR.

So, it seems that in the use case of AQI in India, the CatBoost and random forest algorithms, coupled with SMOTE applied datasets, can provide great results to estimate air quality, which can prompt local and national governments, as well as other civic bodies to act and regulate the air quality. As very evident from the abovementioned metrics, the application of these regression models on the 2015 to 2020 AQI data has been successful in demonstrating that our innovation of using the SMOTE algorithm has paid off well and increased the accuracy values of these regression models. This innovative approach can be applied to future research and its benefits reaped.

References

[1] Predicting Air Quality Index Using Attention Hybrid Deep Learning and ARIMA Models, https://journalofbigdata.springeropen.com/articles/10.1186/s40537-024-00926-5

[2] An Air Quality Index Prediction Model Based on CNN-ILSTM, https://www.nature.com/articles/s41598-022-12355-6

[3] Prediction of Air Quality Index Using Machine Learning Techniques, https://onlinelibrary.wiley.com/doi/10.1155/2023/4916267

[4] Air Quality Index Prediction Using DNN-Markov Modeling, https://www.tandfonline.com/doi/full/10.1080/08839514.2024.2371540

[5] Optimized Air Quality Management Based on Air Quality Index Prediction, https://www.nature.com/articles/s41598-024-68972-w

[6] A Deep Learning Approach for Prediction of Air Quality Index in Smart Cities, https://link.springer.com/article/10.1007/s43621-024-00272-9